

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently Amended) A distributed computer system comprising:
at least two interconnected computers, each of said computers storing:
 - i) component loading code executable to load component program code including one or more callable procedures on said computer, and to store corresponding component procedure interface information for said one or more callable procedures;
 - [[i]]ii) component-~~process~~ program code executable to provide a ~~process~~ program forming part of a distributed software application;
 - [[ii]] iii) event messaging code executable to receive one or more event messages from another of said computers;
 - [[iii]] iv) event reaction rule storage code executable to store, in an updateable store, one or more event reaction rules which include one or more calls to said one or more procedures in said component ~~process-program~~ in reaction to the receipt of said event message;
 - [[iv]] v) event reaction rule interpretation code executable to operate said computer in accordance with said event reaction rules by making one or more calls to one or more procedures included with said event reaction rules;
 - [[v]] vi) event reaction rule modification code executable to allow a user to modify said event reaction rules stored in said updateable store-~~whilst~~ while said component ~~process-program~~ is running and thereby alter the operation of said distributed software application-~~whilst~~ while it is running.
2. (Canceled)

3. (Currently Amended) A distributed computer system according to claim 1 wherein said event reaction rule interpretation code comprises computational reflection code executable to convert method or procedure call data in said event reaction rule into a corresponding method or procedure call for execution.

4. (Currently Amended) A distributed computer system according to claim 1 in which said event messages are structured in accordance with event schema data accessible to each of said computers.

5. (Currently Amended) A distributed computer system according to claim 4 in which said event messages comprise a combination of event data and mark-up data.

6. (Currently Amended) A distributed computer system according to claim 5 in which said event messages are sent as encoded text.

7. (Currently Amended) A distributed computer system according to claim 1 in which an event reaction rule ~~said process-modification code~~ is executable to alter the operation of said distributed software application ~~configure said process~~ by specifying a method or procedure to be called and the parameters to accompany said method or procedure call.

8. (Currently Amended) A distributed computer system according to claim 7 in which said specified method or procedure is running on the other of said computers.

9. (Currently Amended) A distributed computer system according to claim 1 in which said interconnected computers comprise an administration computer having installed thereon graphical user interface code executable to allow an administrator to update said event reaction rules.

10. (Currently Amended) A distributed computer system according to claim 1 in which said event reaction rules specify a method or procedure to be carried out in reaction to the reception of an event message.

11. (Currently Amended) A distributed computer system according to claim 10 in which said event reaction rules further specify a condition to be tested, the carrying out of said action being conditional on said condition being met.

12. (Currently Amended) A distributed computer system according to claim 1 in which each of said computers further stores database management code executable to provide a database store for said rules stored on said computer.

13. (Currently Amended) A distributed computer system according to claim 1 in which each of said computers further stores component-~~process~~ program details including names of one or more procedures or methods provided by said component ~~process~~ program.

14. (Currently Amended) A distributed computer system according to claim 13 in which said component-~~process~~ program details further include names of one or more input parameters to be included with a method call and an indication of the type of those input parameters.

15. (Currently Amended) A distributed computer system according to claim 13 further comprising graphical user interface code executable to enable a user to view said component ~~process~~ program details.

16. (Currently Amended) A method of operating a distributed computer system comprising a plurality of interconnected computers to provide a distributed software application, said method comprising:

operating each of said computers to:

i) load component program code including one or more callable procedures on said computer;

ii) store corresponding component procedure interface information for said one or more callable procedures;

[[i]] iii) execute ~~one or more~~ said component program code to provide a program forming ~~process~~ which form part of a distributed software application;

iv) receive one or more event messages from another of said computers;

~~[[ii]]v)~~ store in an updatable store, one or more event reaction rules, said event reaction rule including a reference one or more calls to said one or more procedures within in said component-process program in reaction to receipt of said event message;

vi) interpret said stored event reaction rules to operate said computer accordance with said event reaction rules; and

~~[[III]}vii)~~ provide a user with an interface allowing a user to modify updating of said event reaction rules stored in said updateable store while said component program is running and thereby alter the operation of said distributed software application while it is running.

~~iv) responsive to the receipt of an event message at said computer, to interpret a corresponding event reaction rule and thereby execute said one or more procedures referred to in said event reaction rule.~~